DOCUMENT RESUME

ED 229 789 CS 207 534

AUTHOR Rodrigues, Raymond J.; Rodrigues, Dawn
TITLE CAI Invention Strategies.
PUB DATE Mar 83
NOTE 17p.; Paper presented at the Annual Meeting of the Conference on College Composition and Communication (34th, Detroit, MI March 17-19, 1983).
PUB TYPE Information Analyses (070) -- Viewpoints (120) -- Speeches/Conference Papers (150)

EDRS PRICE MF01/PC01 Plus Postage.
DESCRIPTORS *Computer Assisted Instruction; *Computer Oriented Programs; *Computer Programs; Elementary Secondary Education; Online Systems; *Prewriting; Problem Solving; Research Needs; *Writing Instruction; *Writing Processes
IDENTIFIERS *Invention (Rhetorical)

ABSTRACT

Prewriting programs using computers fall into two broad categories: interactive and noninteractive. An early example of a noninteractive program is that of Ellen Nold, called "Cinnamon." Its purpose was to present the student with a series of content questions. In answering such questions, the student would be accumulating a set of data that could be incorporated later into a piece of writing. In another noninteractive program, Nold forced her students to consider their audiences. Parallel to Nold's, but somewhat interactive, is a program developed by William Wresch. In his program, Wresch includes specific questions designed to help students structure their writing. An interactive invention program developed by Hugh Burns makes use of the tagmemic matrix of Young, Becker, and Pike; the pentad of Burke; and the enthymeme topoi of Aristotle. A program by Helen Schwartz leads students through prewriting exercises specifically designed to develop their ability to write literary papers. A more recent interactive invention program based upon creative problem-solving techniques makes use of visual synectics, in which a student selects a picture or views a graphic display on the monitor, lists elements of the picture onto the computer, and then relates, through comparisons, how the elements compare to the topic. Along with these developments are the new research questions raised by computer assisted writing instruction. One possible area of research concerns the nature of computer responses as rewards. Another area of investigation might be culturally determined learning styles. Thus, while computer assisted writing programs are exciting tools, they also provide variables for further research. (HOD)

****
* Reproductions supplied by EDRS are the best that can be made *
* from the original document. *
****

U.S. DEPARTMENT OF EDUCATION
NATIONAL INSTITUTE OF EDUCATION
EDUCATIONAL RESOURCES INFORMATION CENTER (ERIC)

X This document has been reproduced as received from the person or organization originating it.

☐ Minor changes have been made to improve reproduction quality.

• Points of view or opinions stated in this document do not necessarily represent official NIE position or policy.

ED229789

CAI INVENTION STRATEGIES

Raymond J. Rodrigues
Department of Curriculum and Instruction

Dawn Rodrigues
Department of English

New Mexico State University

Paper presented at the Conference on College Composition and Communication, Detroit, Michigan, March 17-19, 1983

"PERMISSION TO REPRODUCE THIS MATERIAL HAS BEEN GRANTED BY

Raymond J. Rodrigues

TO THE EDUCATIONAL RESOURCES INFORMATION CENTER (ERIC)."

CS 207 534

For the past ten years or so, researchers in composition have been attempting to develop a number of computer programs, programs based upon accepted invention theories and designed to stimulate students' thinking prior to their actually beginning to write. Many of these programs have been simple cataloging affairs, devices that enable students to list their ideas, to structure them, possibly to revise them, or even to invite the comments of others. Such programs might be said to be essentially substitutes for the instructor, mechanical devices that can be used by students outside the classroom, allowing the instructor additional in-class time for activities related to other stages of the writing process, such as directed revision sessions or peer editing workshops. Other programs have involved students in simulated question and answer sessions, typically some form of heuristic or algorhithmic design. Both approaches have their advantages, ranging from simple economy and efficiency to challenging students to think of ideas that they might never have thought of on their own. Further, students who use computer-assisted invention programs do not have to remember the invention strategy itself--the computer does that; students are free to focus on their emerging topics. As impressive as some of these

programs may be, they are, nevertheless, seeds in a revolution of computer-aided or computer-stimulated instruction, seeds that are only now sending down their roots and beginning to swell with a growth seldom imagined by English teachers.

Following brief descriptions of key programs reported in the literature and of a program we have designed which involves creative problem- solving invention strategies, we will raise some questions researchers need to explore, among these the issue of whether some students work more effectively with certain kinds of invention programs. There may be a relationship between students' learning styles and the effectiveness of computer-assisted writing instruction.

Prewriting programs using computers fall into two broad categories: 1. interactive, most often heuristic, and 2. non-interactive, most often algorhithmic, or catalog programs. Non-interactive programs often ask students questions that simply result in students listing bits of content that later can be integrated into a composition. Another use of non-interactive programs is to have students structure their ideas according to a predetermined plan established by the creator of the particular program. In both instances, the computer program may actually stimulate students to invent new ideas, but such a result is more likely fortuitous. And, of course, there are non-interactive programs that lead students both to develop content and to structure their writing.

Interactive programs, from our perspective, are the more exciting of the available invention via computer formats. In such programs, the computer, as a result of the God-like whim of the original creator, actually takes on a persona, a machine-like character that talks to the student, encourages the student, chides the student, jokes with the student, helps the student with more information--in short, that pretends to actually be a teacher who cares what the student has to say. The parallels between such programs and actual instructors are often frighteningly real: the computer is loaded with a series of stock responses, sometimes very Rogerian or non-directive in nature, and, regardless of what the student says, the computer is able to draw from its data bank and randomly throw out a question or stimulus that will keep the student talking, keep the student writing, keep the student thinking. Granted, students can fool the computer. Depending upon the complexity of the programs, students could put garbage in to the computer storage file with impunity, or the students could be caught and chastized for attempting such faithless subterfuge. Since computers can recognize syntactic categories more easily than they can recognize semantic categories, students might still type in syntactically correct semantic nonsense, but we have to assume that most students will approach their tasks with good faith.

An early example of a non-interactive program is that of Ellen Nold, reported in CCC in 1975.[1] Nold called her program "Cinnamon," and its purpose was to present the student with a series of content questions. In answering such questions, the

student would be accumulating a set of data that could be later incorporated into a piece of writing. For example, one set of questions dealt with the arbitrariness of naming. The program began with the question, "Why do you suppose the woman is always the one to change her name?" The student could respond with anything--from a sarcastic "who cares?" or a naive "because that's the way it always has been" to a serious argument against the practice. The computer, of course, is totally ignorant, and proceeds as though it has not heard a word the student has said, which is, in fact, what has happened. "Our tradition has it that a woman belongs to her husband. The name indicates a shift in the power relationship." (The parallel with some teachers is indeed frightening.) Then, before the student can register a complaint, the computer proceeds: "Speaking of tradition, why do you suppose God had Adam and Eve name all of the animals, instead of doing it himself?"

In another non-interactive program, Nold forced her students to consider their audiences. This program, "Arrowroot," begins, "Use two or three words to state the subject of your paper." Student response number one is then recorded by the computer. Next, the computer asks, "What is the main point you will make about (response number 1)?" The computer records student response number two. Then the computer asks, "Who is your audience? Whom are you trying to persuade or convince?" The student types in response number three. Now the computer asks, "What does (response number three) already know about (response number one)?" In such a manner, the program continues, asking questions about the attitude that the writer can anticipate of

the audience, specific supports that would help the argument, and the best order for those specific supports. Built into such a program would be a system, or network, for other students or the instructor to make comments upon what the writer has already said.

Parallel to Nold's, but somewhat interactive, is a program developed and reported by William Wresch.[2] In his program, Wresch includes specific questions designed to help students structure their writing. The program begins by asking students to state their general topics. From there, the program moves on to ask for a more narrow topic. Regardless of how narrow the original topic has been, the computer still has a stock response, "That sounds like a good topic, but maybe you could come up with an even better subject if you got more specific....Do you think you could make your subject more specific? Why not reenter your subject and then hit the return key again?" This, too, sounds disturbingly like an English teacher who has learned the job by rote. But then Wresch's program asks students to list the major attributes of the topic. Here the students have a choice: they may choose to answer some questions about the history of the topic, the appearance of the topic, the causes of the topic, or the function of the topic. Having given the computer a number of responses to the program's questions, the students then see the computer list those responses, something few teachers would take the time to do. Then the program asks the students to rank the three most important responses. After the students do so, the program asks the students to type in as many statements as they

can about each of the three most important responses. Having gone through this routine for all of the major responses, the program at last prints out the total responses of the students. What results is a bare outline of the traditional five-paragraph essay. At this point, the computer prints out an explanation of what has just transpired. Throughout, the program has adopted an encouraging and humble tone--"I'm just a computer and so don't know.' From this, the student should infer that the student is indeed the expert on this topic--the student is the expert on content, the tutor (or computer program) is merely the facilitator. That's an approach that more instructors might adopt themselves.

One of the most complex, yet delightful, approaches in interactive invention programs is that of Hugh Burns.[3] Burns' program made use of the tagmemic matrix of Young, Becker, and Pike; the pentad of Burke; and the enthymeme topoi of Aristotle. Burns designed his program to study such questions as which heuristic would most effectively elicit sustained responses (the topoi) and which would be least effective (the pentad) and which would produce the most propositions (the pentad). In terms of pure quality and quantity of ideas, either heuristic approach seemed to work well.

Burns' program took students through anywhere from thirty to sixty minutes of questioning and then provided students with a printout of all their responses, allowing them to go away with a body of data that they could rework into an essay. The tone of the program is chatty and encouraging:

YOUR TURN, KAREN.

I WANT TO SHOW HOW SISTER CARRIE ALLOWED DREISER TO GROW, TO CONFRONT COMMON VALUES. IT IS A MATURE WORK.

GOOD, KAREN. ADD TO YOUR RESPONSE NOW.

HURSTWOOD GROWS IN THE OTHER WAY. THERE ARE NEGATIVE PASSAGES, AREN'T THERE?

YES, THAT SEEMS OKAY TO ME.

ME, TOO.

AH, SHORT AND SWEET. PLEASE ELABORATE A LITTLE.

WELL, CHARACTERS EITHER GET WORSE OR BETTER; THEY EITHER PASS TO A BETTER UNDERSTANDING OR NOT.

TERRIFIC, KAREN!
ANYTHING ELSE?

The program urges the student on through positive responses, such as, "Super!" "That's an interesting reason. Anything else?" or "Why not? Remember we're exploring for answers." As in Wresch's program, the computer is modest. Asked a question by the student, the program might respond, "I can't pretend to know _____, but you've asked an important question. Remember it." In fact, the student may have asked a stupid question, but the computer is non-judgmental. Programmed to respond in good faith, the computer allows the student to respond in kind.

Burns' program also serves as a tutor, reminding students of lessons studied in class or read in a book, giving examples, and asking the student at several points whether that student would like more help. It stands today as an outstanding example of what can be accomplished in an open-ended, interactive tutorial that leads students through an invention process and serves as a secretary, taking everything down that the student types in.

A variation upon the themes set by Burns and Wresch has been the program of Helen Schwartz.[4] She set out to develop a program that could help students through a literature class, that could lead them through prewriting exercises specifically designed to develop their ability to write literary papers. Here we find no casual tone, treating the student as friend or fellow human. Here we plunge into the serious business of analyzing literature, and so the questions of her heuristic tend to be rather straightforward. In the first part of her program, Schwartz provides a series of questions that the student answers in what Schwartz refers to as a "solo mode." For example, in developing a thesis about characterization,

NAME A FICTIONAL CHARACTER X IN A LITERARY WORK.

SATAN IN PARADISE LOST

DESCRIBE THE CHARACTER X BY COMPLETING THE FOLLOWING: X IS_____.

SATAN IS TRICKY.

PROVIDE EVIDENCE TO SHOW THAT X IS Y: WHAT DOES SATAN DO THAT SHOWS SATAN IS TRICKY?

HE ENTERS THE BODY OF A SERPENT TO DISGUISE HIMSELF....

WHAT DOES SATAN SAY THAT SHOWS SATAN IS TRICKY

HE TELLS EVE LIES ABOUT THE TREE OF WISDOM TO GET HER TO EAT FROM IT

Schwartz' program then goes on to ask about how others react to Satan's trickiness, how Satan compares to others, and what the 3rd person narrator (if there is one) has to say about all this. Then the program asks for contrary evidence, and in view of that evidence, asks whether the student want to revise what has been

written so far.

In handling the potential problem of students writing garbage, whether intentionally or through some form of ignorance, Schwartz shifts the students to a "network mode," in which other writers, including the instructor, can comment upon the ideas generated by the original writer. The student can also read the comments of other writers about the same subject, if that is what they chose. In this respect, Schwartz' program opens the way for more authentic interaction, not writer to computer to writer, but writer to writer, with the computer merely being a convenient intermediary.

Much earlier, Ellen Nold[5] had developed a program that resulted in a very simple poem. The program asked students to fill in slots following simple directions. By printing just the student responses, the computer generated a student-created "poem." In the slot/filler manner, Schwartz' "Saga" program built upon Nold's poem program, applying the technique in a productive way to composition.

Because different students may benefit from alternate invention approaches, we have developed interactive invention programs based upon creative problem-solving techniques. These programs employ the open-ended inquiry approach facilitated by such heuristic programs as that of Burns, the possible structure that might result from programs such as Wresch has created, and the algorhithmic creativity of programs such as Nold has pioneered. Among the techniques we have been working on are

visual synectics, in which a student selects a picture or views a graphic display on the monitor, lists elements of the picture into the computer, and then relates, through comparisons, how the elements compare to the topic. For example:

WHAT WOULD YOU LIKE TO WRITE ABOUT, RAY?

COMPUTER ASSISTED INSTRUCTION

The student then receives instructions on how to proceed, selects a picture, and tells the computer he or she is ready to continue.

DESCRIBE WHAT YOU SEE IN THE PICTURE. TYPE A LIST OF WHAT YOU SEE, STARTING WITH NUMBER 1.

1. PILLOWS
2. RUGS
3. A TABLE
4. A SCULPTURE
5. A CHAIR

TRY TO FIND SOME OTHER THINGS IN THE PICTURE.

6. A TRAY
7. A CABINET
8. A PLANT

OKAY, RAY, NOW REMEMBER THAT YOU WANT TO WRITE ABOUT: COMPUTER ASSISTED INSTRUCTION. I WANT YOU TO THINK ABOUT HOW COMPUTER ASSISTED INSTRUCTION COMPARES TO PILLOWS.

THEY BOTH MAKE OUR LIVES A GREAT DEAL SOFTER.

NOW THINK ABOUT HOW COMPUTER ASSISTED INSTRUCTION COMPARE(S) TO RUGS.

THEY BOTH COVER A LARGE AREA.

HOW DOES COMPUTER ASSISTED INSTRUCTION SEEM SIMILAR TO A TABLE?

THEY BOTH PROVIDE SUPPORT FOR MANY THINGS.

The program continues through this process until it comes to the end of the list. Then it lists all of the statements that the

student has typed into the computer and, one by one, as in some of the other programs reported, asks the student to think through each statement, listing supporting ideas for each. Finally, the computer prints out all of the ideas the student has had, providing the student a copy to study while developing the paper.

We are also in the process of developing programs that make use of focused objects, relational algorithms, simple analogies, and matrix analyses.

These creative problem solving strategies are designed to stimulate student thinking, not to structure or to judge it. If anything, it is our hope that students may discover ideas they had never conceived of. Robert Frost argued that the tight structure of a sonnet, or of any other traditional or non-traditional highly structured poem, actually freed the mind by forcing it to think of possible ideas that free verse never could. So it is with creative problem solving techniques. They are highly artificial, highly structured attempts to lead the writer to freedom of ideas, to force the writer to think in ways that casual questions could not, and possibly that even commonly accepted heuristic designs could not. What remains for us now is to determine whether that is so.

The questions raised by computer-aided writing instruction, whether about invention techniques, wordprocessing capability,[6] or the facility of flexible editing programs such as "Writers' Workbench,"[7] open research areas never before available.One possible area of research concerns the nature of computer responses as rewards. Do the pre-programmed computer responses

inhibit or facilitate the invention processes of writers? Burns found that "CAI could be programmed which encouraged both growth in the number and the sophistication of ideas."[8] Yet, Kenneth McGraw[9] has raised some questions about the nature of reward, indicating that reward may facilitate learning in essentially algorithmic situations, learning situations in which paths to solutions are straightforward and well-mapped. Heuristic approaches, in which the learner must essentially discover not only answers, but also paths--or algorithms--to the answers may be detrimental if rewards are tied into the process. From what we have observed, computer assisted instruction may be reward enough in and of itself, so that students work through the computer-assisted process simply because of its innate fascination for students. But we do not know whether that fascination will remain once computers are as common as paper and pen. Given the current high motivation that attends computer assisted instruction, heuristic approaches may well work as well as algorithmic approaches.

Culturally-determined learning styles also present a broad area for investigation. Ramirez and Castaneda,[10] for example, have investigated the nature of field-dependent learners vs. field-independent learners. Field-dependent learners depend upon external variables to support and motivate their efforts in an educational setting. A pat on the shoulder, a smile, a reference to how proud the learner's family would be of the progress being made, all may help. The field-independent learner, on the other hand, relies less upon external cues and more upon internal

verification of progress. Will the same differences apply when students confront the computer? Will interactive programs provide more support for the field-dependent learner than non-interactive programs? And what of the tone of those programs? Will conforting, soothing words urge students to even greater production, even if programmed to be randomly selected as computer reponses and even if the program keeps cautioning the student that "after all, I am only a computer!" Will the field-independent learner react negatively to such computerized ammenities and produce more when the program is open-ended and barely structured?

Investigators have categorized learning styles in a variety of schemes.[11] Gregorc[12], for example, has produced a scheme of learning styles which allows us to develop research questions along another line of thinking. Gregorc classifies learners into combinations developed along polar extremities: random vs. sequential and concrete vs. abstract. (Most university professors happen to fall into the abstract-random thinker category, but their students may run the full range.) Does a prewriting or invention program facilitate the writing of concrete sequential learners more than abstract sequential or concrete random or abstract random learners? Is the concrete nature of a highly structured printout of more value than just the process of interacting with a computer? Can random learners develop their writing through computer programs with greater facility if they are in a writing lab working by themselves than if they are in a class with an instructor? And what exactly is the best combination of instructor-aided writing instruction and

computer-aided writing instruction? Does computer-aided instruction have more to offer students in the prewriting phase or in the writing/wordprocessing phase or in the editing phase? Or do any of these differences matter? And if they do, how do we determine those combinations?

The advent of computer-assisted composition programs has opened a realm of promising research. Universities are rushing to the computer because it offers a hopeful means for handling those time-consuming tasks that currently prohibit us from effectively dealing with individual students. In composition, editing programs, because of their efficiency, have a clear value for both instructor and student. Prewriting programs, however, require that computers interact with human minds. So, while those interactions make computer-assisted prewriting programs exciting tools, we also know that they provide a variable in human learning we have yet to understand, much yet master. as both instructor and student.

NOTES

1. Ellen Nold, "Fear and Trembling: The Humanist Approaches the Computer," College Composition and Communication, 26 (October 1975), 269-273.

2. William Wresch, "Prewriting, Writing, and Editing by Computer," paper presented at the thirty-third annual meeting of the College Conference on College Composition and Communication, San Francisco, March 18-20, 1982. 14pp. [ED213045]

3. Hugh L. Burns, Jr., "A Writer's Tool: Computing as a Mode of Inventing," paper presented at the New York College English Association Conference, October 3-4, 1980. 11 pp. [ED193693]

4. Helen Schwartz, "A Computer Program for Invention and Feedback," paper presented at the thirty-third annual meeting of the Conference on College Composition and Communication, San Francisco, March 1982. 10 pp. [ED 214177]

5. Nold, Ibid.

6. Kate Kiefer, "The Development of Computer-Assisted Instruction in Composition," and Charles Smith, "Writer's Workbench: Applications in the Classroom," paper presented at the thirty-fourth annual meeting of the College Conference on Composition and Communication, Detroit, March 1982.

7. Lillian S. Bridwell, "The Effects of the Use of Word Processors on Written Texts and Composing Processes: Case Studies," paper presented at the seventy-second Annual Convention of the National Council of Teachers of English, Washington, D.C., November 19-24, 1982.

8. Hugh L. Burns, Jr., Stimulating Rhetorical Invention in English Composition Through Computer-Assisted Instruction. Austion: University of Texas, 1979. 68 pp. [ED188245]

9. Kenneth McGraw, "The Detrimental Effects of Reward on Performance: A Literature Review and a Prediction," in Mark R. Lepper and David Greene (eds.), The Hidden Costs of Reward, New York: Lawrence Erlbaum Association: Hillsdale, N.J.,1978.

10. Manuel Ramirez III and Alfredo Castaneda, Cultural Democracy, Bicognitive Development, and Education.New York: Academic Press, 1974.

11. See, for example, National Association of Secondary School Principals, Student Learning Styles and Brain Behavior, Reston Va.: NASSP, 1982.

12. A self-report instrument allowing students to categorize themselves according to four combinations of learning preference dualities is The Gregorc Style Delineator, available from Anthony F. Gregorc, Doubleday Road, Columbia, Connecticut 06237.